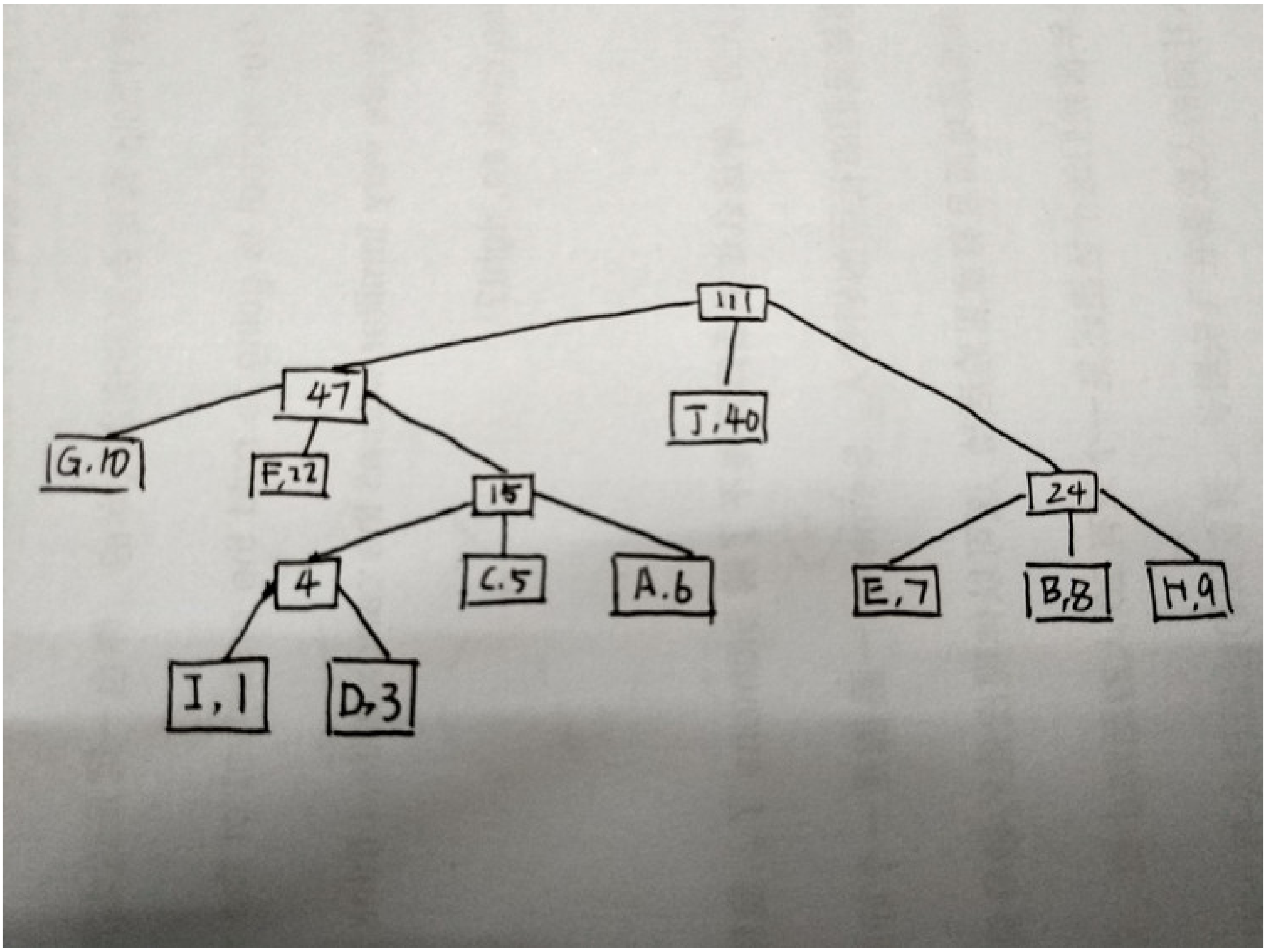
数据结构与算法

第五章 习题

1

> 已知某电文中共出现了10种不同的字母（A, B, C, D, E, F, G, H, I, J），每种字母出现的次数分别为6、8、5、3、7、22、10、9、1、40，现在对这段电文用三进制进行非定长前缀编码（码字由0、1、2组成），请问Huffman电文编码总长度至少有多少位？相应于等长编码，只需要多少空间？请画出相应编码方案的图示（不需要画中间过程，只需要最终的结果）。



字符	A	B	C	D	E	F	G	H	I	J
编码	022	21	021	0201	20	01	00	22	0200	1

总长度：201 节省空间：132

2

> 证明：判断以下三种情况是否成立，并给出证明，对于不成立的，需要给出反例：
>
> **已知先序遍历序列和中序遍历序列可以确定唯一一棵二叉树。**

利用数学归纳法，1个节点的树只有一种形态显然满足。现假设对于节点个数$\leq K$的树都满足这一性质，则当节点个数$N = K + 1$时：

设先序遍历为$x_1, x_2, ..., x_N$, 中序遍历为$z_1, z_2, ..., z_N$， 则一定$\exists i, s.t.\ x_1 = z_i$，即$x_2, x_3, ..., x_i$和$z_1, z_2, ..., z_{i-1}$为左子树的先序遍历和中序遍历，$x_{i+1}, xi + 2, ..., x_N$和$z_{i+1}, z_{i+2}, ..., z_N$为右子树的先序遍历和中序遍历。左右子树的节点个数都$\leq K$,由归纳假设，我们可以唯一的确定左右子树，而根也是确定的，故我们可以唯一的确定这棵树。

即$N = K + 1$也满足这一性质，故所有的二叉树都满足这一性质。

> **已知中序遍历序列和后序遍历序列可以确定唯一一棵二叉树。**

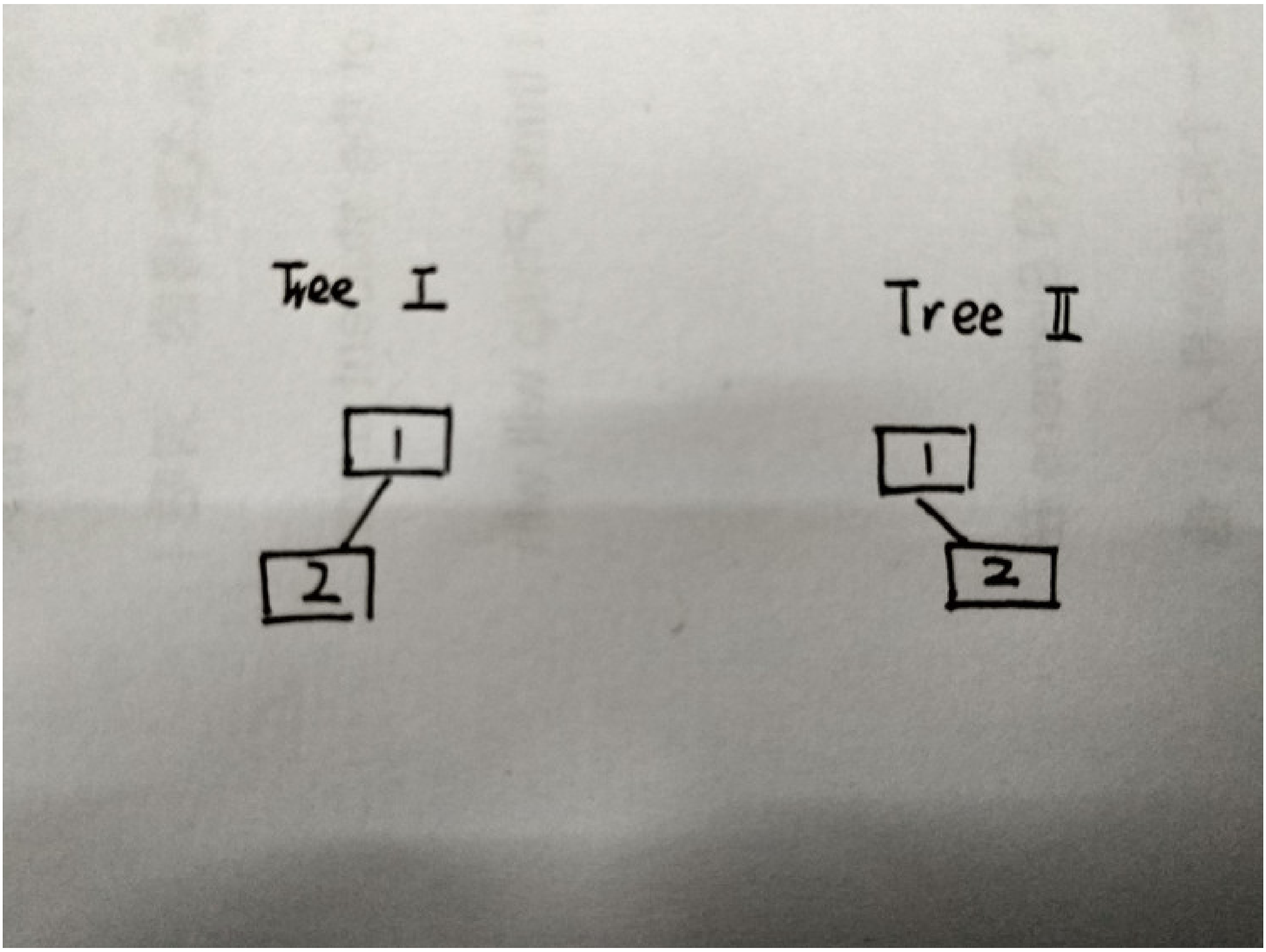
利用数学归纳法，1个节点的树只有一种形态显然满足。现假设对于节点个数$\leq K$的树都满足这一性质，则当节点个数$N = K + 1$时：

设中序遍历为$z_1, z_2, ..., z_N$, 后序遍历为$h_1, h_2, ..., h_N$， 则一定$\exists i, s.t.\ z_i = h_N$，即$z_1, x_2, ..., x_{i-1}$和$h_1, h_2, ..., h_{i-1}$为左子树的中序遍历和后序遍历，$z_{i+1}, z_{i+2}, ..., z_n$和$h_i, h_{i+1}, ..., h_{n-1}$为右子树的中序遍历和后序遍历。左右子树的节点个数都$\leq K$,由归纳假设，我们可以唯一的确定左右子树，而根也是确定的，故我们可以唯一的确定这棵树。

即$N = K + 1$也满足这一性质，故所有的二叉树都满足这一性质。

> **已知先序遍历序列和后序遍历序列可以确定唯一一棵二叉树。**

显然不成立。

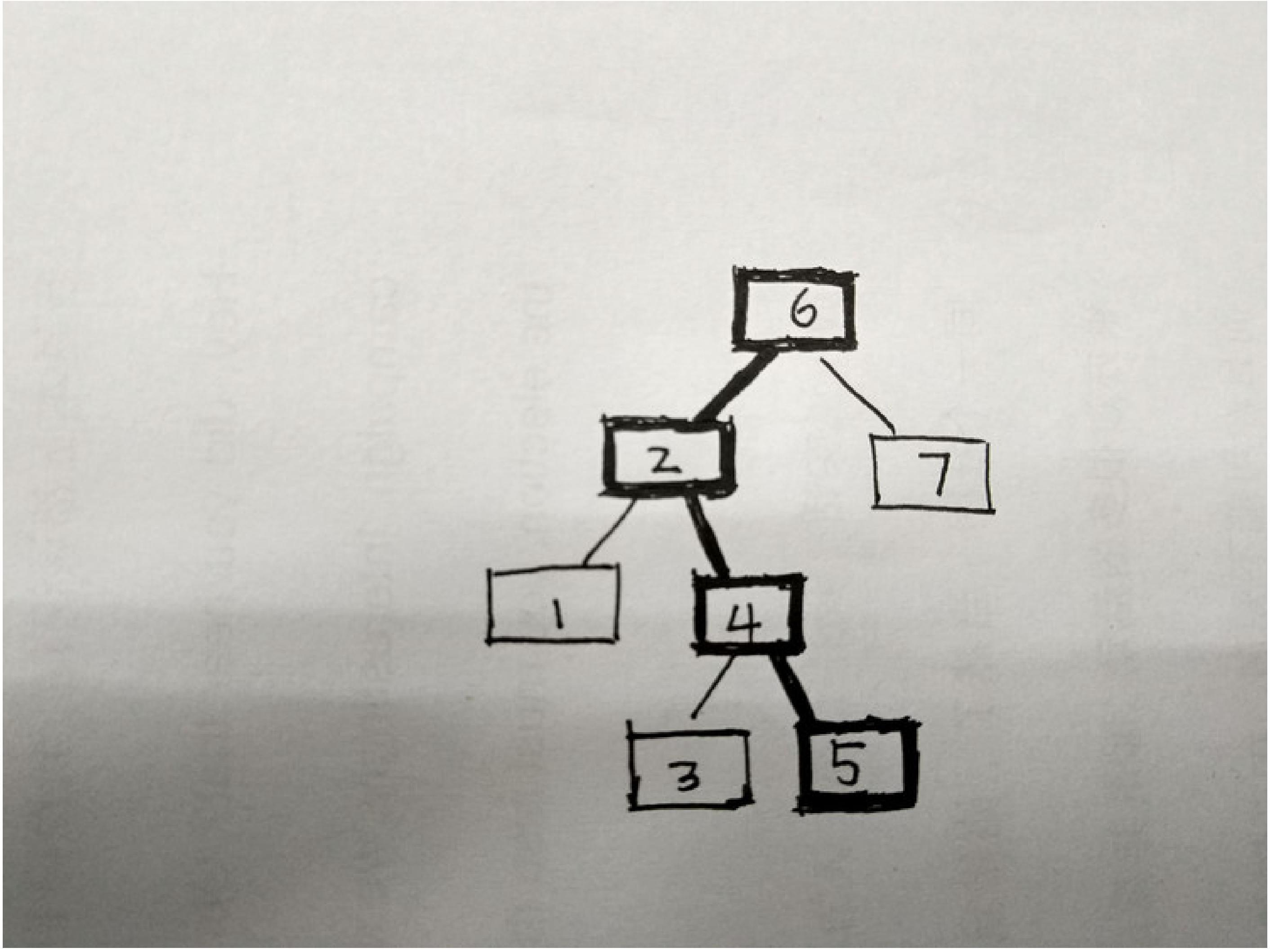


先序遍历和后序遍历都是1, 2和2, 1

3

> 1. 在一棵表示有序集S的二叉搜索树中，任意一条从根到叶子结点的路径将S分为3个部分：在该路径左边结点中的元素组成的集合S1；在该路径上的结点中的元素组成的集合S2；在该路径右边结点中的元素组成的集合S3。S = S1∪S2∪S3。若对于任意的a∈S1，b∈S2，c∈S3是否总有a≤b≤c？为什么

显然不是。



加粗的是路径，但是$3 > 2$.

4

> 下面的算法是使用非递归的方法中序遍历二叉树。请填充算法中的空格，使其成为完整的算法。

```
element.tag = Left;
aStack.top().tag == Right
pointer = aStack.top().pointer()->rightchild();
```

5

> 下面的算法是从根节点开始广度优先遍历二叉树。若二叉树有n个结点，设遍历过程中的队列的最大长度是f(t)，t可以是任意一棵二叉树。当t为完全二叉树时，f(t)取得最大值，这种说法对吗？如果正确，请给出证明；如果错误，请举出反例。（完全二叉树的定义：只有最后一层的节点是可能为空的，且最后一层中左边是满的）

先证明节点个数为N的树T，$\max(f(T)) \le \frac{N+1}{2}$。一个节点的树$\max(f) = 1$显然满足性质，现假设对于节点个数$\le K$的树都满足这一性质，则当节点个数$N = K + 1$时：

当队列中有根节点时长度是1，当队列中没有根节点时，队列中的节点可以分为在左子树中的节点和在右子树中的节点两种，这两种形态分别会出现在对左右子树进行广度优先遍历的队列中。故
$f(N) \leq \max(f(K) + f(N-1-K)) = \max(\frac{K+1}{2} + \frac{N-1-K+1}{2}) \leq \frac{N+1}{2}$。

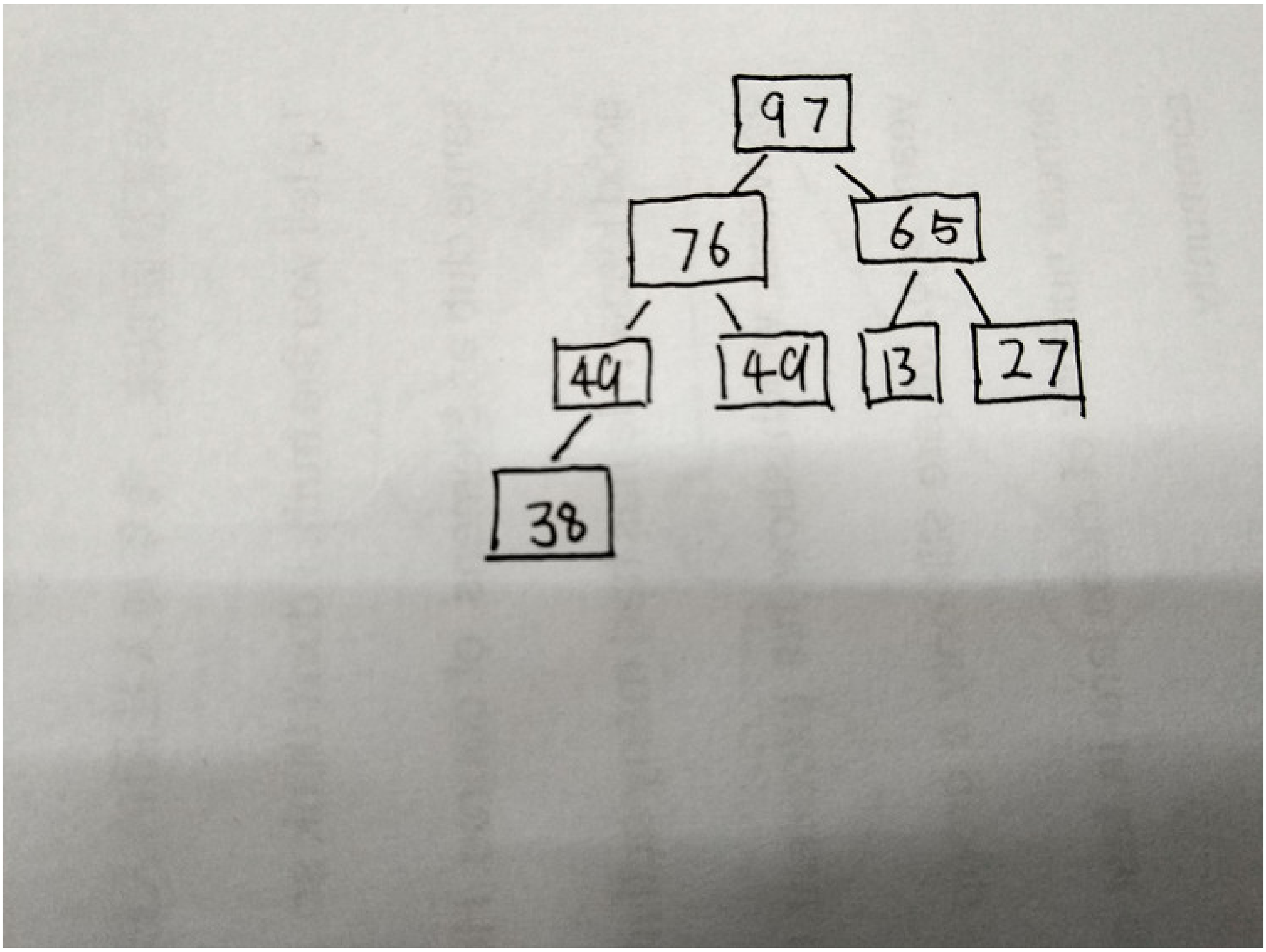
即$N = K + 1$也满足这一性质，故所有的二叉树都满足这一性质。

接下来证明完全二叉树T的$f(T) \geq \frac{N+1}{2}$, 由于N个节点的二叉树有$\frac{N+1}{2}$个叶子节点，而完全二叉树的叶子节点满足**只有最后一层的节点是可能为空的，且最后一层中左边是满的**，即叶子节点的BFS序号连续且在最后。当队列中最后一个有叶子节点的分支节点出队列后，队列中有且仅有全部的叶子节点，此时队列大小为$\frac{N+1}{2}$。

综上，当T为完全二叉树时$f(T)$取到最大的值。

6

> 初始关键码序列为{49, 38, 27, 49, 76, 13, 65, 97}，试给出用筛选法所建立的最大堆，并写出其相应的序列。在建堆过程中，移位次数是多少?



序列：$\{97, 76, 65, 49, 49, 13, 27, 38\}$

位移次数：6